**Państwowa Wyższa Szkoła Informatyki i Przedsiębiorczości w Łomży**

**Wydział Informatyki i Nauk o Żywności**

**Kierunek studiów: Informatyka II stopień**



**Anita Żochowska** 8444

**Piotr Ołtarzewski** 7820

**Mateusz Wojnarowski** 8439

**Patryk Zabłocki** 8441

**Sprawozdanie z projektu**

**Temat projektu: Atlas Drinków**

**Przedmiot: Projekt grupowy**

**Prowadzący: dr inż. Janusz Rafałko**

**Łomża 27.11.2021**

1. **Wstęp**

Tematem projektu jest Atlas drinków. Ma on za zadanie wyświetlanie dodanych przez użytkowników przepisów na drinki oraz ich zdjęcia. Użytkownik podczas dodawania będzie mógł dodać zdjęcia, dokładny przepis, zawartość procentową drinku. Dodane dane będą przechowywane w bazie danych. Dodatkową funkcją aplikacji będzie możliwość obliczenia czasu potrzebnego do wytrzeźwienia po wypiciu przyrządzonego drinku. Aplikacja dla użytkownika będzie w formie webowej oraz mobilnej co sprawi że użytkownik będzie mógł korzystać niezależnie czy będzie w domu czy w innym miejscu.

1. **Wymagania funkcjonalne**

W przypadku tworzonego projektu mamy do czynienia z dwoma aplikacjami webową oraz mobilną dla których zostaną przedstawione oddzielne wymagania.

**2.1 Aplikacja webowa**

* Przeglądanie i wyszukiwanie drinków przez użytkownika
* Dodawanie propozycji przepisu na drink przez użytkownika (przepis wyświetla się wraz z innymi po zaakceptowaniu przepisu przez moderatora)
* Panel moderatora do akceptacji lub odrzucenia przepisu zaproponowanego przez użytkownika
* Kalkulator promili

**2.2 Aplikacja mobilna**

* Przeglądanie i wyszukiwanie drinków przez użytkownika
* Dodawanie propozycji przepisu na drink przez użytkownika (przepis wyświetla się w raz z innymi po zaakceptowaniu przepisu przez moderatora)
* Kalkulator promili

1. **Wymagania niefunkcjonalne**

W przypadku tworzonego projektu mamy do czynienia z dwoma aplikacjami webową oraz mobilną dla których zostaną przedstawione oddzielne wymagania niefunkcjonalne.

**3.1 Aplikacja webowa**

* Poprawne działanie po uruchomieniu
* Poprawne wyświetlanie zawartości
* Musi działać na wszystkich systemach operacyjnych
* Aplikacja musi działać płynnie
* Nie może powodować krytycznych błędów
* Nie może wykorzystywać 100% zasobów
* Krótki czas wykonywania zapytań do bazy danych

**3.2 Aplikacja mobilna**

* Poprawne działanie po uruchomieniu
* Poprawne wyświetlanie zawartości
* Musi działać na większości wersji systemu android
* Aplikacja musi działać płynnie
* Nie może powodować krytycznych błędów
* Nie może powodować zawieszanie się androida
* Nie może wykorzystywać 100% zasobów
* Krótki czas wykonywania zapytań do bazy danych

1. **Opis biznesowy**

Celem projektu jest stworzenie aplikacji która będzie pełniła rolę książki z przepisami dla drinków. Będzie ona umożliwiała dodawanie najróżniejszych przepisów drinków prze użytkowników. Dodane przepisy będą potem wyświetlane w aplikacji webowej lub mobilnej, co sprawi że zniknie potrzeba uczenia się na pamięć wielu różnych przepisów, a każdy kto będzie korzystał z tej aplikacji będzie miał możliwość poznania nowych przepisów na drinki w intuicyjnej i przyjaznej dla użytkownika aplikacji. Oprócz wspomnianych wyżej funkcjonalności nasza aplikacja będzie posiadała opcję wyliczenia czasu potrzebnego do wytrzeźwienia po wypiciu danego drinku, wystarczy że użytkownik wypełni wymagane pola odpowiednimi danym, a aplikacja pokaże mu czas potrzebny do wytrzeźwienia. Dzięki aplikacji mobilnej użytkownicy będą mieli możliwość dostępu do książki z przepisami w każdym miejscu co sprawi że świetnie ona się sprawi poza domem oraz na imprezach okolicznościowych. Aplikacja webowa oraz mobilna będą posiadały bardzo przyjazny oraz intuicyjny interfejs dzięki czemu użytkownik bez problemu odnajdzie treści których potrzebuje lub doda nowe przepisy. Nasz projekt jest skierowany w głównej mierze dla osób które ukończyły 18 rok życia z racji tego że opiera się na tematyce alkoholu.

1. **Metodyka**

Model kaskadowy (waterfall) - polega on na wykonywaniu podstawowych czynności jako odrębnych faz projektowych, w porządku jeden po drugim. Każda czynność to kolejny schodek (kaskada):

1. Planowanie systemu (w tym specyfikacja wymagań)
2. Analiza systemu (w tym analiza wymagań i studium wykonalności)
3. Projekt systemu (poszczególnych struktur itp.)
4. Implementacja (wytworzenie kodu)
5. Testowanie (poszczególnych elementów systemu oraz elementów połączonych w całość)
6. Wdrożenie i pielęgnacja powstałego systemu.

Jeśli któraś z faz zwróci niesatysfakcjonujący produkt cofamy się wykonując kolejne iteracje aż do momentu kiedy otrzymamy satysfakcjonujący produkt na końcu schodków.

Metodyka ta została wybrana ponieważ najbardziej pasuje do naszego projektu.

1. **Podział pracy**

* Aplikacja webowa – Patryk Zabłocki
* Baza danych – Patryk Zabłocki
* Aplikacja mobilna - Piotr Ołtarzewski / Mateusz Wojnarowski
* Przygotowanie elementów graficznych dla aplikacji – Anita Żochowska
* Tworzenie dokumentacji – Anita Żochowska / Mateusz Wojnarowski / Piotr Ołtarzewski
* Testowanie – wszyscy członkowie grupy

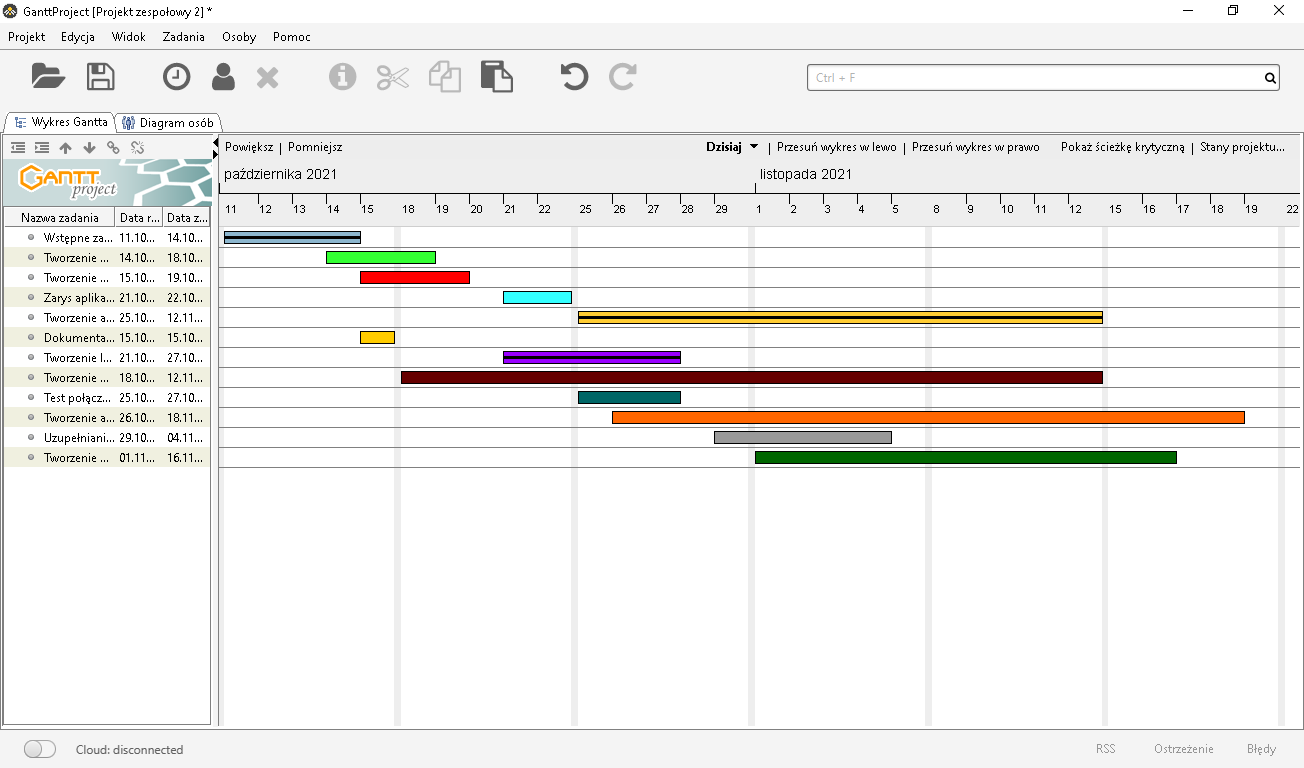
1. **Harmonogram**

|  |  |
| --- | --- |
| Deadline | Nazwa elementu projektu |
| 9.10.2021 | Określenie tematu pracy oraz podziału zadań |
| 16.10.2021 | Analiza literatury i dostępnych rozwiązań |
| 23.10.2021 | Wykonanie wstępnej dokumentacji projektu (**Opis biznesowy projektu, wymagania funkcjonalne i niefunkcjonalne, technologia, metodyka, podział pracy, harmonogram**) |
| 30.10.2021 | Opracowanie algorytmów działania aplikacji, struktury bazy danych, protokołów wymiany informacji między podsystemami, interfejsu użytkownika |
| 6.11.2021 | Wykonanie diagramów UML |
| 13.11.2021 | Implementacja aplikacji |
| 20.11.2021 | Przetestowanie i naprawianie błędów, dokumentacja techniczna projektu |
| 27.11.2021 | Gotowy projekt |

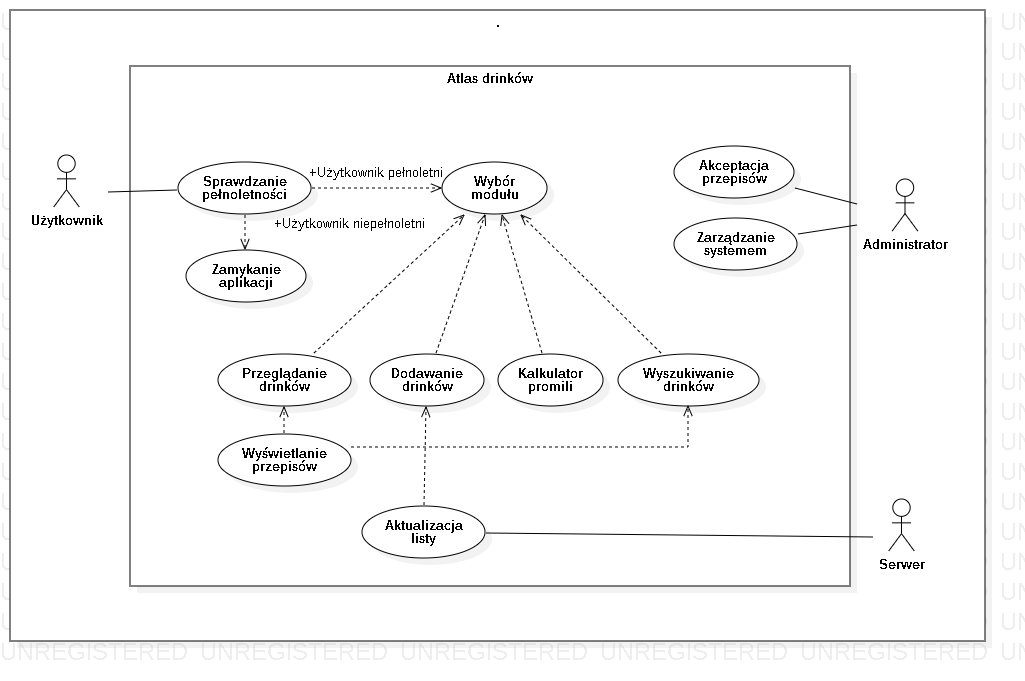
**8. Technologia**

* Aplikacja webowa:
  + React.js + TypeScript
  + Visual Studio Code
* Aplikacja mobilna
  + Java
  + Android Studio
* Baza danych
  + Restdb.io

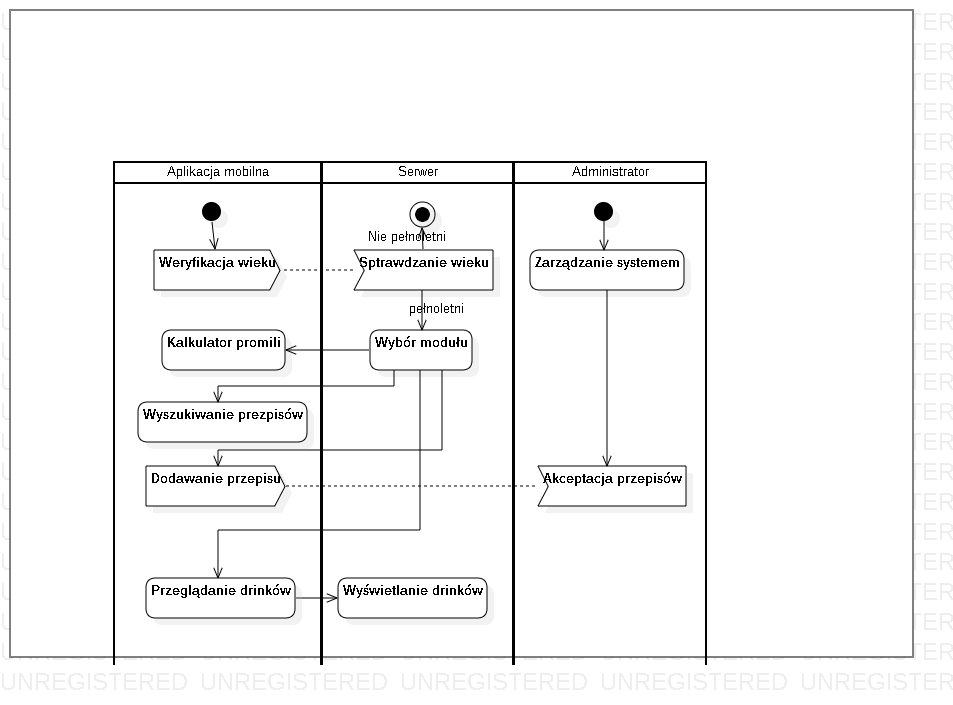
1. **Diagram Gantta**



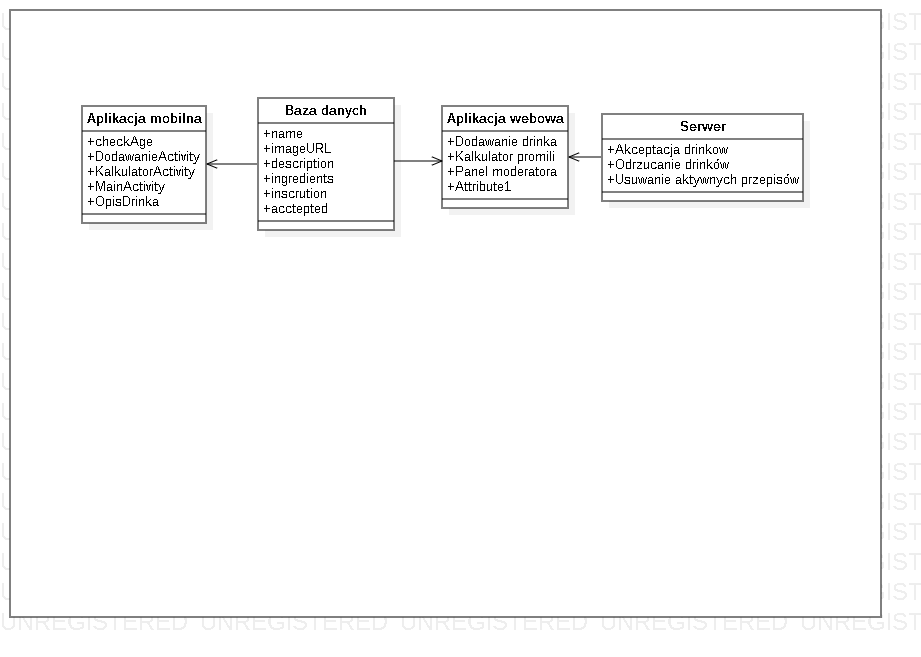
1. **Diagram przypadków użycia**



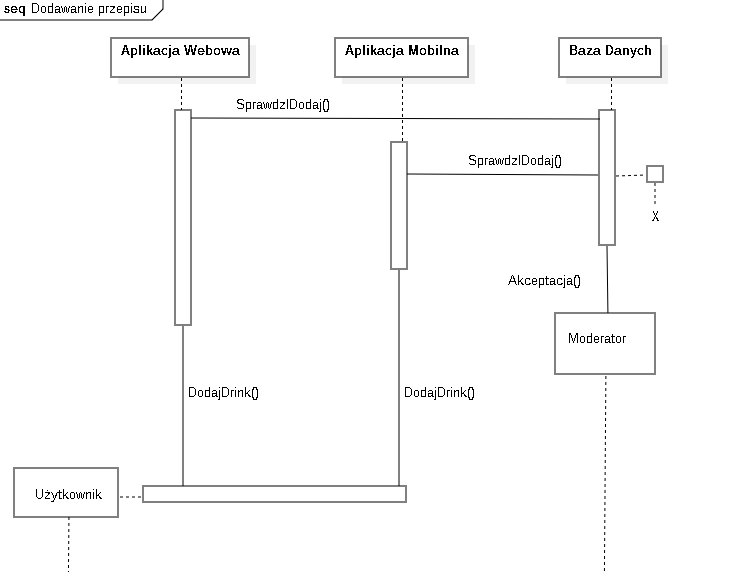
1. **Diagram aktywności**



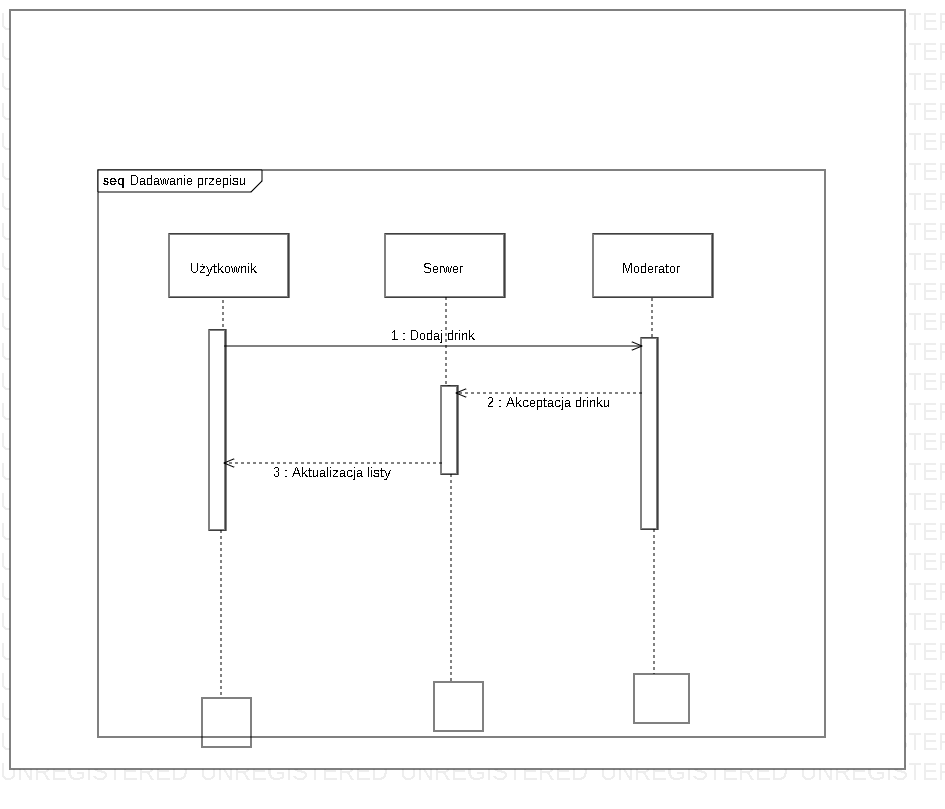
1. **Diagram klas**



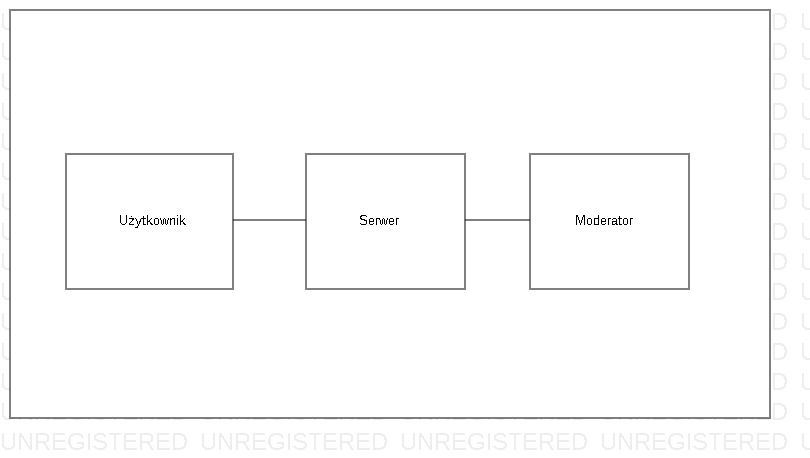
1. **Diagramy sekwencji**



1. **Diagramy komunikacji**



1. **Diagram wdrożenia**



1. **Kod źródłowy aplikacji**

* **Kod aplikacji wejściowej**

package com.example.drinki;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.text.Editable;

import android.text.TextWatcher;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.EditText;

import android.widget.ListView;

import android.widget.SearchView;

import android.widget.Toast;

import java.util.Arrays;

public class MainActivity extends AppCompatActivity{

ListView listadrinkow;

EditText Szukaj;

ArrayAdapter<String> nazwyAdapter;

//public String nazwy[];

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

listadrinkow = findViewById(R.id.drinki\_list);

Szukaj = findViewById(R.id.searchFilter);

Szukaj.setText(null);

//Odebranie bazy danych z drinkami

Intent intent = getIntent();

String[] nazwy = intent.getStringArrayExtra("nazwy");

String[] opis = intent.getStringArrayExtra("opis");

String[] skladniki = intent.getStringArrayExtra("skladniki");

String[] instrukcja = intent.getStringArrayExtra("instrukcja");

String[] linkiPhoto = intent.getStringArrayExtra("linki");

//Wrzucenie nazw drinków do listview, a ona jest interaktywna

nazwyAdapter = new ArrayAdapter<>(MainActivity.this, android.R.layout.simple\_list\_item\_1, nazwy);

listadrinkow.setAdapter(nazwyAdapter);

Szukaj.addTextChangedListener(new TextWatcher() {

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {

MainActivity.this.nazwyAdapter.getFilter().filter(s);

}

@Override

public void afterTextChanged(Editable s) {

}

});

listadrinkow.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

String nazwa = parent.getItemAtPosition(position).toString();

Intent przegladDrinka = new Intent(MainActivity.this, OpisDrinka.class);

//Wysłanie wszystkich informacji o wybranym drinku

przegladDrinka.putExtra("nazwa", nazwy[(int)id]);

przegladDrinka.putExtra("opis", opis[(int)id]);

przegladDrinka.putExtra("skladniki", skladniki[(int)id]);

przegladDrinka.putExtra("instrukcja", instrukcja[(int)id]);

przegladDrinka.putExtra("linkPhoto", linkiPhoto[(int)id]);

startActivity(przegladDrinka);

}

});

}

}

* **Kod dodawania przepisu**

package com.example.drinki;

import android.content.Intent;

import android.os.Bundle;

import android.os.StrictMode;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.io.BufferedInputStream;

import java.io.BufferedOutputStream;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.io.OutputStream;

import java.io.OutputStreamWriter;

import java.net.MalformedURLException;

import java.net.URL;

import java.net.URLEncoder;

import javax.net.ssl.HttpsURLConnection;

public class DodawanieActivity extends AppCompatActivity {

Button dodaj\_button;

EditText przepis\_view;

Button usun\_button;

EditText jak\_zrobic\_view;

EditText Nazwa\_view;

EditText Opis\_view;

EditText Link\_view;

public String nazwy, opis, skladniki, metoda, linkiPhoto;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_dodawanie);

Intent intent = getIntent();

dodaj\_button =(Button)findViewById(R.id.dodaj\_but);

usun\_button =(Button)findViewById(R.id.usun\_but);

przepis\_view = (EditText)findViewById(R.id.przepis\_textview);

jak\_zrobic\_view = (EditText)findViewById(R.id.jak\_zrobic\_textview);

Nazwa\_view = (EditText)findViewById(R.id.Nazwa\_textView);

Opis\_view = (EditText)findViewById(R.id.Opis\_textView);

Link\_view = (EditText)findViewById(R.id.link\_przepisView);

StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().permitAll().build(); //dodałem

StrictMode.setThreadPolicy(policy);

dodaj\_button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Thread thread2 = new Thread(new Runnable() {

@Override

public void run() {

try {

URL koktajle = null;

try {

koktajle = new URL("https://atlasdrinkow-9348.restdb.io/rest/drinki");

} catch (MalformedURLException e) {

System.out.println("Blad jest tutaj: " + e.toString() + " !!!!!!!!!!!!!");

e.printStackTrace();

}

HttpsURLConnection myConnection = null;

try {

myConnection = (HttpsURLConnection) koktajle.openConnection();

myConnection.setRequestProperty("User-Agent", "my-restdb-app");

myConnection.setRequestProperty("Accept", "application/json");

myConnection.setRequestProperty("x-apikey", "6171abab8597142da17458d7");

myConnection.setRequestMethod("POST");

nazwy = Nazwa\_view.getText().toString();

opis = Opis\_view.getText().toString();

skladniki = przepis\_view.getText().toString();

metoda = jak\_zrobic\_view.getText().toString();

linkiPhoto = Link\_view.getText().toString();

String data = URLEncoder.encode("name", "UTF-8")

+ "=" + URLEncoder.encode(nazwy, "UTF-8");

data += "&" + URLEncoder.encode("description", "UTF-8") + "="

+ URLEncoder.encode(opis, "UTF-8");

data += "&" + URLEncoder.encode("ingredients", "UTF-8")

+ "=" + URLEncoder.encode(skladniki, "UTF-8");

data += "&" + URLEncoder.encode("instructions", "UTF-8")

+ "=" + URLEncoder.encode(metoda, "UTF-8");

data += "&" + URLEncoder.encode("imageUrl", "UTF-8")

+ "=" + URLEncoder.encode(linkiPhoto, "UTF-8");

data += "&" + URLEncoder.encode("accepted", "UTF-8")

+ "=" + URLEncoder.encode("false", "UTF-8");

String text = "";

Log.d("data", data + "");

BufferedReader reader = null;

//if (myConnection.getResponseCode() == 200) { // z jakiegoś powodu bez tego sprawdzenia dziala prawidlowo, prawdopodobnie baza odsyla kod z zakresu 200-299

myConnection.setDoOutput(true);

myConnection.setChunkedStreamingMode(0);

OutputStreamWriter wr = new OutputStreamWriter(myConnection.getOutputStream());

wr.write(data);

wr.flush();

Log.d("msg", "Wysłane");

InputStream in = new BufferedInputStream(myConnection.getInputStream());

reader = new BufferedReader(new InputStreamReader(myConnection.getInputStream()));

String line = "";

StringBuffer buffer = new StringBuffer();

while ((line = reader.readLine()) != null) {

buffer.append(line);

}

Log.d("msg","Baza odeslala: " + buffer.toString());

} catch (IOException e) {

Log.d("data", e + "");

e.printStackTrace();

}

} catch (Exception e) {

e.printStackTrace();

Log.d("data", e + "");

}

}

});

thread2.start();

Toast.makeText(DodawanieActivity.this, "Dodano nowy drink!", Toast.LENGTH\_SHORT).show();

}

});

usun\_button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Nazwa\_view.setText("");

Opis\_view.setText("");

przepis\_view.setText("");

jak\_zrobic\_view.setText("");

Link\_view.setText("");

}

});

}

}

* **Kod kalkulatora promili**

package com.example.drinki;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.ProgressBar;

import android.widget.RadioButton;

import android.widget.RadioGroup;

import android.widget.SeekBar;

import android.widget.Switch;

import android.widget.TextView;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import androidx.appcompat.widget.Toolbar;

import java.util.ArrayList;

public class KalkulatorActivity extends AppCompatActivity {

public static TextView waga;

public static Switch plec;

private static RadioGroup drinkSizeGroup;

private static RadioButton drinkRozmiar;

private static SeekBar procentAlkocholu;

private static TextView alcPerDisplay;

private static TextView bacPoziom;

private static double bacLevelDisplay = 0.00;

private static double finalBACPoziom = 0.00;

private static TextView finalowyWynik;

private static int minAlkochol = 5;

private static int obecnyAlkochol = 5;

private static int maxAlkochol = 100;

private static int stepAlkochol = 5;

private static Double r = 0.0;

private static int W = 0;

private static RadioButton radio30ml ;

private static RadioButton radio60ml;

private static RadioButton radio266ml;

private static String stringWaga;

private static ArrayList<Integer> PochlonientyAlkochol = new ArrayList<>();

private ProgressBar progressBar;

private static int progress;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activirt\_kalkulatorv2);

Intent intent = getIntent();

progressBar = findViewById(R.id.progressBarBACLevel);

bacPoziom = findViewById(R.id.textViewBACLevel);

drinkSizeGroup = findViewById(R.id.radioGroupDrinkSize);

radio30ml = findViewById(R.id.radioButton30ml);

radio60ml = findViewById(R.id.radioButton60ml);

radio266ml = findViewById(R.id.radioButton266ml);

waga = findViewById(R.id.editTextWaga);

plec = findViewById(R.id.switchPlec);

finalowyWynik = findViewById(R.id.textViewResult);

finalBACPoziom = Math.round(bacLevelDisplay \* 100.0) / 100.0;

bacPoziom.setText(Double.toString(finalBACPoziom));

progress = (int) (finalBACPoziom \* 100);

progressBar.setMax(100);

progressBar.setProgress(progress);

seekbar();

setResultColor();

findViewById(R.id.button2).setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

saveButtonClick();

}

});

drinkSizeGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(RadioGroup group, int checkedId) {

drinkRozmiar = findViewById(checkedId);

}

});

findViewById(R.id.buttonDodajDrink).setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if(waga.getText().toString().length() <=0) {

waga.setError("Wprowadz wage");

} else {

if (W == 0 && r == 0.0) {

saveButtonClick();

}

String stringDrinkSize;

if (drinkRozmiar == null) {

stringDrinkSize = "30 ml";

} else {

stringDrinkSize = (String) drinkRozmiar.getText();

}

stringDrinkSize = stringDrinkSize.replace(" ml", "");

PochlonientyAlkochol.add(Integer.parseInt(stringDrinkSize) \* obecnyAlkochol);

bacLevelDisplay = bacLevelDisplay + ((PochlonientyAlkochol.get(PochlonientyAlkochol.size() - 1) \* 6.24) / (W \* r \* 100));

finalBACPoziom = Math.round(bacLevelDisplay \* 100.0) / 100.0;

progress = (int) (finalBACPoziom \* 100);

progressBar.setProgress(progress);

bacPoziom.setText(Double.toString(finalBACPoziom));

setResultColor();

enableDisable();

}

}

});

findViewById(R.id.buttonReset).setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

bacLevelDisplay = 0.00;

finalBACPoziom = Math.round(bacLevelDisplay \* 100.0) / 100.0;

bacPoziom.setText(Double.toString(finalBACPoziom));

waga.setText("");

W = 0;

r = 0.0;

plec.setChecked(false);

setResultColor();

enableDisable();

//reset do 0

progressBar.setProgress(0);

radio30ml.setChecked(true);

radio60ml.setChecked(true);

radio266ml.setChecked(true);

//reset do 0

procentAlkocholu = findViewById(R.id.seekBarAlcohol);

procentAlkocholu.setProgress(0);

}

});

}

public void saveButtonClick() {

if(waga.getText().toString().length() <=0) {

waga.setError("Wprowadz wage");

}else {

stringWaga = "" + waga.getText();

W = Integer.parseInt(stringWaga);

if (plec.isChecked()) {

r = 0.68;

} else {

r = 0.55;

}

if (bacLevelDisplay != 0.00) {

bacLevelDisplay = 0.00;

calculateBACLevel();

setResultColor();

}

enableDisable();

}

}

public void enableDisable() {

if(finalBACPoziom >= 1.0){

findViewById(R.id.buttonDodajDrink).setEnabled(false);

Toast.makeText(KalkulatorActivity.this, "Tobie juz wystarczy", Toast.LENGTH\_LONG).show();

}else{

findViewById(R.id.buttonDodajDrink).setEnabled(true);

}

}

public void setResultColor() {

if(finalBACPoziom <= 0.50) {

finalowyWynik.setText("Bezpieczny");

} else if(finalBACPoziom >= 0.55 && finalBACPoziom <= 0.75) {

finalowyWynik.setText("Uwazaj");

}else if(finalBACPoziom >= 0.95) {

finalowyWynik.setText("Przekroczyles limit");

}

}

public void calculateBACLevel(){

String stringDrinkSize;

if (drinkRozmiar == null) {

stringDrinkSize = "30 ml";

} else {

stringDrinkSize = (String) drinkRozmiar.getText();

}

stringDrinkSize = stringDrinkSize.replace(" ml", "");

for (int i = 0; i < PochlonientyAlkochol.size(); i++) {

bacLevelDisplay = bacLevelDisplay + ((PochlonientyAlkochol.get(i) \* 6.24) / (W \* r \* 100));

}

finalBACPoziom = Math.round(bacLevelDisplay \* 100.0) / 100.0;

progress = (int) (finalBACPoziom \* 100);

progressBar.setProgress(progress);

bacPoziom.setText(Double.toString(finalBACPoziom));

enableDisable();

}

public void seekbar (){

procentAlkocholu = findViewById(R.id.seekBarAlcohol);

alcPerDisplay = findViewById(R.id.textViewSeekbarProgress);

procentAlkocholu.setMax(maxAlkochol - minAlkochol);

procentAlkocholu.incrementProgressBy(stepAlkochol);

procentAlkocholu.setProgress(0);

alcPerDisplay.setText(obecnyAlkochol + "%");

procentAlkocholu.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {

@Override

public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {

progress = progress / 5;

progress = progress \* 5;

obecnyAlkochol = progress + minAlkochol;

alcPerDisplay.setText(obecnyAlkochol + "%");

}

@Override

public void onStartTrackingTouch(SeekBar seekBar) {

}

@Override

public void onStopTrackingTouch(SeekBar seekBar) {

alcPerDisplay.setText(obecnyAlkochol + "%");

}

});

}

}

* **Kod sprawdzania pełnoletności**

package com.example.drinki;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.io.BufferedInputStream;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.net.MalformedURLException;

import java.net.URL;

import javax.net.ssl.HttpsURLConnection;

public class CheckAge extends AppCompatActivity implements View.OnClickListener {

Button tak\_buttoon;

Button nie\_button;

public String[] nazwy, opis, skladniki, metoda\_Przyzadzenia, linkiPhoto;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_check\_age);

tak\_buttoon =(Button)findViewById(R.id.tak\_but);

tak\_buttoon.setOnClickListener(this);

nie\_button=(Button) findViewById(R.id.nie\_but);

nie\_button.setOnClickListener(this);

}

@Override

public void onClick(View v) {

if(v == tak\_buttoon){

Thread thread= new Thread(new Runnable() {

@Override

public void run() {

URL koktajle = null;

try {

koktajle = new URL("https://atlasdrinkow-9348.restdb.io/rest/drinki?q={\"accepted\":true}");

} catch (MalformedURLException e) {

System.out.println("Blad jest tutaj: " + e.toString() + " !!!!!!!!!!!!!");

e.printStackTrace();

}

System.out.println("URL przyjety !!!!!!!!!!!");

HttpsURLConnection myConnection = null;

try {

myConnection = (HttpsURLConnection) koktajle.openConnection();

myConnection.setRequestProperty("User-Agent", "my-restdb-app");

myConnection.setRequestProperty("Accept", "application/json");

myConnection.setRequestProperty("x-apikey", "6171abab8597142da17458d7");

if(myConnection.getResponseCode() == 200)

{

InputStream in = new BufferedInputStream(myConnection.getInputStream());

BufferedReader reader = new BufferedReader(new InputStreamReader(in, "UTF-8"));

String line="";

StringBuffer buffer = new StringBuffer();

while ((line = reader.readLine())!=null){

buffer.append(line);

}

JSONArray jsonArray = new JSONArray(buffer.toString());

nazwy = new String[jsonArray.length()];

opis = new String[jsonArray.length()];

skladniki = new String[jsonArray.length()];

metoda\_Przyzadzenia = new String[jsonArray.length()];

linkiPhoto = new String[jsonArray.length()];

DrinksContener drinksContener;

for(int i = 0; i< jsonArray.length(); i++)

{

JSONObject jsonObject = jsonArray.getJSONObject(i);

String name = jsonObject.getString("name");

nazwy[i] = name;

String description = jsonObject.getString("description");

opis[i] = description;

String ingredients = jsonObject.getString("ingredients");

skladniki[i] = ingredients;

String instrukcja = jsonObject.getString("instructions");

metoda\_Przyzadzenia[i] = instrukcja;

String adresZdjecia = jsonObject.getString("imageUrl");

linkiPhoto[i] = adresZdjecia;

}

Intent lista = new Intent(CheckAge.this,MainActivity.class);

lista.putExtra("nazwy", nazwy);

lista.putExtra("opis", opis);

lista.putExtra("skladniki", skladniki);

lista.putExtra("instrukcja", metoda\_Przyzadzenia);

lista.putExtra("linki",linkiPhoto);

startActivity(lista);

}

else

{

}

} catch (IOException e) {

e.printStackTrace();

} catch (JSONException e) {

e.printStackTrace();

}

}

});

thread.start();

}

else

Toast.makeText(CheckAge.this, "Jestes za mlody", Toast.LENGTH\_SHORT).show();

}

}

1. **Podsumowanie projektu**

Pomysł na projekt został wybrany i poparty przez całą grupę. Chcieliśmy by nasz projekt był ciekawy i wyjątkowy dlatego wybraliśmy tematykę alkoholu. W projekcie napotkaliśmy wiele problemów takich jak brak doświadczenia wprogramowaniu przy pomocy Javy i korzystania z programu android studio, w którym została stworzona nasza aplikacja mobilna. Sama aplikacja posiadała wiele błędów których naprawienie było bardzo wymagające i nadal posiada wiele bugów. Nasza aplikacja pierwotnie miała być bardziej rozwinięta jednak z braku aktualnej wiedzy zrezygnowaliśmy z wielu pomysłów i aplikacja stała się bardzo przystępna i prosta. Jednym z pomysłów był bardziej przejrzysty interfejs, panel logowania czy panel moderatora nie tylko z aplikacji webowej a i mobilnej. Aplikacja webowa zaś wyszła bardzo dobrze i nie było z nią większych problemów tak jak i połączenie jej z bazą danych. W ogólnym rozrachunku to był nasz pierwszy projekt z użyciem aplikacji mobilnej i z zdobytą przez ten projekt wiedzą kolejne nie powinny nam sprawić większych problemów. Z samego projektu jesteśmy zadowoleni i wynieśliśmy z niego dużo wiedzy.

1. **Załączniki**

<https://restdb.io> – baza danych

<http://koktajle-drinki.pl> - grafiki przedstawiające drinki

<http://koktajle-drinki.pl> – strona z przepisami

<https://www.mojbar.pl/przepisy-na-drinki> - inna strona z przepisami

<https://courses.cs.duke.edu/fall05/cps001/labs/lab4.html> - inspiracja wzoru kalkulatora

1. **Wykorzystane źródła**

<https://www.youtube.com>

<https://stackoverflow.com>

<https://github.com>

<https://pl.wikipedia.org>

<https://www.microsoft.com/pl-pl>